大规模的RDF数据存储技术综述*

何少鹏 黎建辉 沈志宏 周园春

(中国科学院计算机网络信息中心 北京 100190)

摘要:随着互联网上数据大规模的增长以及语义网的发展,如何存储大规模RDF成为了当前普遍关注的问题。本文对语义网中大规模RDF存储系统的研究现状与进展进行了分析,分别介绍了在RDF存储系统中的存储组织和查询优化以及现有的一些大规模RDF存储系统解决方案,重在对大规模的RDF数据存储技术研究的主流方法和前沿进展进行分析,最后对大规模RDF存储系统存在的一些问题进行了讨论并展望了未来的发展方向。

关键词:RDF框架,语义网,存储系统,云计算

Overview of the Storage Technology for Large - scale RDF Data

HE Shaopeng, LI Jianhui, SHEN Zhihong, ZHOU Yuanchun

(Computer Network Information Center, Chinese Academy of Sciences, Beijing, 100190, China)

Abstract: With the massive growth of the data on the Internet as well as the development of the Semantic Web, the problem of Storing massive RDF has become more and more concerned. This paper surveys and analysis the development of Large - scale RDF Storage System, including the storage pattern, query optimization, and some solutions for Large - scale RDF Storage Systems. This paper aims to make a summary and analysis for storage technology on Large - scale RDF data. At last, this paper gives some problems for current studies and proposes some research directions.

keywords: RDF, semantic Web, storage system, cloud computing

1 引言

随着互联网的高速发展,Web上的数据呈现出指数级的增长,对大规模的数据存储系统提出了要求。另一方面,由于传统的数据没有包含足够的语义信息,无法让计算机理解各种数据的含义,这就要求数据要规范化和智能化。语义网[1]可以看做是一种数据的网络,它使用规范化的格式来表达数据,使得数据可以表达语义信息,RDF(Resource Description Framework)[2]作为语义网中资源的描述标准,在语义网中已经占据了重要的地位。RDF格式的数据大量涌现,使得RDF数据存储面临着巨大的挑战,大规模RDF存储系统成为当前迫切的问题。

近些年来,RDF存储系统作为语义网的一个重要的研究方向已经吸引了大量的研究者,国外的研究机构包括美国南加州大学、德克萨斯大学、宾夕法尼亚大学、惠普实验室、德国Karlsruhe大学以及人工智能研究中心、韩国大学等等,国内的研究机构有北京大学、华中科技大学、上海交通大学等等。早期的研究主要集中在RDF数据存储模式包括如何用关系型数据库来存储[3]RDF、SPARQL查询分析、推理规则以及综合

本文于2012-09-05收到。

*本课题是中国科学院战略性先导科技专项"面向感知中国的新一代信息技术研究"子课题"数据资源中心及其关键技术"(编号:XDA06010202)。

系统的研发。但是,这种存储方式一方面需要额外的 RDF 向关系型转换过程,另一方面关系型数据库有限存储规模成为 RDF 大规模存储的约束。随着大规模 RDF 存储系统的迫切需求,很多机构已经开始了其他存储格式方面的研究,比如南加州大学的 RDFPeers[4],德克萨斯大学的基于 Hadoop 和 MapReduce 的 RDF 存储和检索[5],上海交通大学的 HadoopRDF 系统[6]等。本文在现有的大规模 RDF 存储系统的基础上,对大规模的 RDF 数据存储技术进行了分析和综述。

2　语义网和 RDF

万维网创始人 Tim Berners - Lee 于 2001 年提出的关于下一代互联网的设想[7]即语义网,语义网的主要任务就是让计算机理解 Web 上的大量数据的含义,使得数据更加便于处理和查找,从而为人们提供更加智能化的服务。其主要思想就是对互联网上的任意资源,进行结构化的描述并引入语义,使得计算机可以理解互联网上的信息。

在语义网的体系架构中[8],RDF(Resource Description Framework)是 W3C(World Wide Web Consortium)提出的描述 Web 信息的通用语言。RDF 作为一种 Web 上的知识表示语言,它提出用一个简单的模型来表示任意类型的数据,这个模型采用三元组(S,P,O)来描述 Web 信息,其中 S 代表主语(Subject),P 代表谓语(Predicate),O 代表宾语(Object)。从图的角度来看,该模型由节点以及节点之间的边组成的,节点分别表示主语 S 和宾语 O,连接节点 S 和节点 O 的边表示谓语 P,这样就可以用节点来表示 Web 上的资源,用边来表示资源的属性。因此,这个数据模型可以方便的描述对象(或者资源) 以及它们之间关系。

W3C 在 RDF 数据查询语言上,先后提出了 OWL - QL[9],RQL[10],RDQL[11],SPARQL[12]等语言。其中,SPARQL 已逐渐成为一种流行的标准化的语言。SPARQL 包含 4 种查询方式,分别为 SELECT、CONSTRUCT、DESCRIBE 和 ASK。其中,SELECT 是最为常见的查询方式,类似于 SQL 中的 SELECT。SPARQL 支持各种平台和语言,编写更复杂的查询。为了进一步提炼查询的结果,SPARQL 拥有 DISTINCT、LIMIT、OFFSET 和 ORDERBY 等关键字,它们的操作或多或少地与 SQL 中的对应命令有些类似。

目前为止,国际上已经提出很多种语义网中数据的描述语言和查询语言,但是广泛被使用的还是 RDF 描述框架和 SPARQL 查询语言,因此,随着语义网的不断的发展,已经出现了大量的 RDF 数据和相应的存储系统以及 SPARQL 查询系统。

3　RDF 存储方法分析

目前已经出现了很多 RDF 存储管理系统,按照存储介质不同可以分为:基于内存、基于文件系统和基于关系数据库的存储方法[13]。

3.1　基于内存的存储方法

该方法主要是将 RDF 数据加载到内存中,然后在内存中进行相应的处理,这种方式的特点是处理速度快,查询效率高,但是规模有限。其中 Sesame 系统框架[14]就提供了基于内存的存储系统的实现;华中科技大学语义网与知识管理研发组开发的 DBLink 系统[15],实现了基于内存的大规模 RDF 三元组数据存储,DBLink 系统使用高效的压缩技术,有效存储和处理了 3.3 Billion 三元组的数据,并提供了 SPARQL 和 DQE 的查询接口进行数据检索和关系查询。

3.2　基于文件系统的方法

该方法主要是将 RDF 数据以文件的形式存储起来,这种方式比较简单,容易实现。其中较为简单的一种方式就是以 XML 格式的树形结构来组织文件,但一个 RDF 图可以序列化为多个不同的 XML 文档,并且查询依赖于 RDF 图的 XML 表示,所以这种存储方式不便于对 RDF 数据查询,尤其当文件比较大时,系统的效率非常低。另外,一些系统基于文件系统建立了 Native 的 RDF 存储管理机制,通过设计专门的存储模式来提高 RDF 数据的存储规模和查询效率。Kowari[16]是一个典型的 Native RDF 存储管理系统,它在存储设

计中利用数据冗余提高查询效率,存储的数据量是原数据量的 6 倍。System Π[17]是东南大学和清华大学联合开发的一个 Native RDF 存储系统,它采用了超图来表示 RDF 数据模型,有效的避免了数据模型转换所消耗的代价。

3.3 基于关系型数据库的存储方法

该方法主要是利用现有的成熟的关系型数据库(比如 SqlServer、MySql、Oracle 等)来存储 RDF 数据。大多数 RDF 应用系统都是采用关系型数据库进行存储的,如 Jena[18],3store[19],Rstar[20]等。采用关系型数据库,可以利用其成熟的组织管理和事务控制以及其针对查询的一些优化,而且数据库管理系统提供了标准查询接口,为 RDF 查询的实现屏蔽了许多底层的操作,因此基于传统关系数据库是一种很好的方案。但是由于 RDF 采用的是三元组来表示数据,而关系型数据库使用的是二维表,二者在模式上存在着差异,因此要将 RDF 存储到关系型数据库中,首先要解决的问题就是如何将 RDF 格式的数据映射为关系型数据。目前已经有很多方式被提出来,常见的基于数据库的 RDF 存储的具体形式有基于水平表[21]、基于垂直表[22]、基于属性建表[21]、混合模式[23]等。

总之这三种方式各有优劣,基于内存的存储方式存储速度较快,却受内存大小的限制,只适应于小规模 RDF 数据。基于文件系统的存储方式实现简单,但对于大规模 RDF 数据,查询效率低。基于关系型数据库,技术成熟,应用广泛,无需重新开发,目前大多数都是用这种方式。但是对于大规模的 RDF 数据存储时,这三种方式都显得捉襟见肘,虽然关系型数据库可以承载的数据量比较大,但是扩展性不好,无法适应目前 RDF 数据的大规模的增长,而且大量的 RDF 数据存储在关系型数据库中,对于查询来说,其低效率是一个必须面对的问题。为此,很多研究机构已经提出了一些大规模 RDF 数据存储的解决方案。

4 大规模 RDF 存储系统分析

传统集中式的 RDF 存储系统无法适应爆炸式增长的海量 RDF 数据,因此人们将目光投向了分布式领域。由于分布式系统同时具备海量存储和并行计算的能力,因此成为解决大规模 RDF 存储的一个适宜途径。基于此,目前已经提出了很多的相应的方案和系统,大致分为采用传统分布式技术和采用云计算技术这两种。

4.1 采用传统分布式技术

分布式系统随着计算机技术的发展已经取得了很大的进步,当前在 RDF 领域已经出现了一些基于通用分布式技术的存储方案与一些为 RDF 数据存储专有优化过的集群方案。下面将对部分已有的系统进行相应的介绍。

4.1.1 RDFPeer

RDFPeers[4]是一个基于 P2P(Peer - to - Peer)技术的分布式 RDF 存储系统。它是由 RDF 三元组存储节点构成的构建在 MAAN(Multi - Attribute Addressable Network)之上的一个 P2P 网络,该系统将一个 RDF 三元组存放在网络中三个不同的节点上,这三个节点分别存放该 RDF 三元组的主语、谓语、宾语,并且采用全局的 Hash 函数进行索引,因此所有节点都知道哪个节点存放了要查询的三元组的值。RDFPeers 的系统结构图 1。从图中可以看出,RDFPeers 集群中的每个节点由如下几个模块组成:MAAN Network Layer、RDF Triple Loader、RDF Triple Storage、Native Query Reslover、RDQL Translator。在 RDF 查询上,RDFPeers 根据查询选择响应

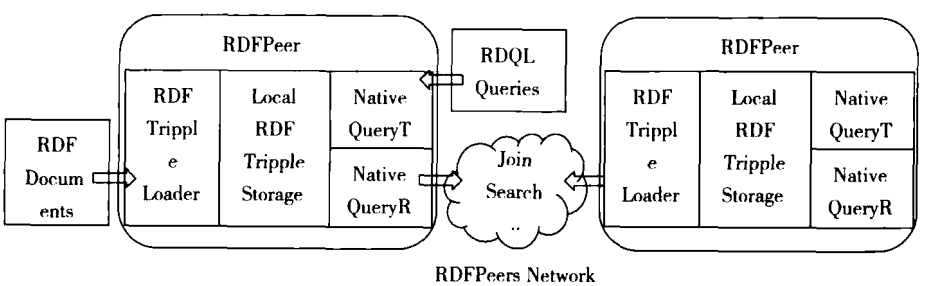

图 1 RDFPeers 系统结构

节点,并通过节点上的心 RDQL Translator 与 Native Query Resolver 进行处理。

4.1.2　4store

4store[24]是一个集群式 RDF 存储系统。在 RDF 的数据存储上,4store 支持存放 RDF 四元组,即 Model、Subject、Predicate、Object。在数据的划分上,4store 为每一个 Subject 分配一个 RID,而后根据 RID 将 RDF 数据划分在不同的 Segment 上并存放在不同的节点上。4store 的节点被分为 Processing 节点和 Storage 节点,这些节点可以被部署在不同的机器上,当然也可以部署到同一台机器上。4store 的系统结构如图 2 所示,从图 2 可以看出,节点中的 RDF 数据存放在三个不同的索引当中 P Index、M Index、R Index。其中 P Index 用于索引 Predicate,M Index 用于索引 Model,R Index 用来索引 RID。

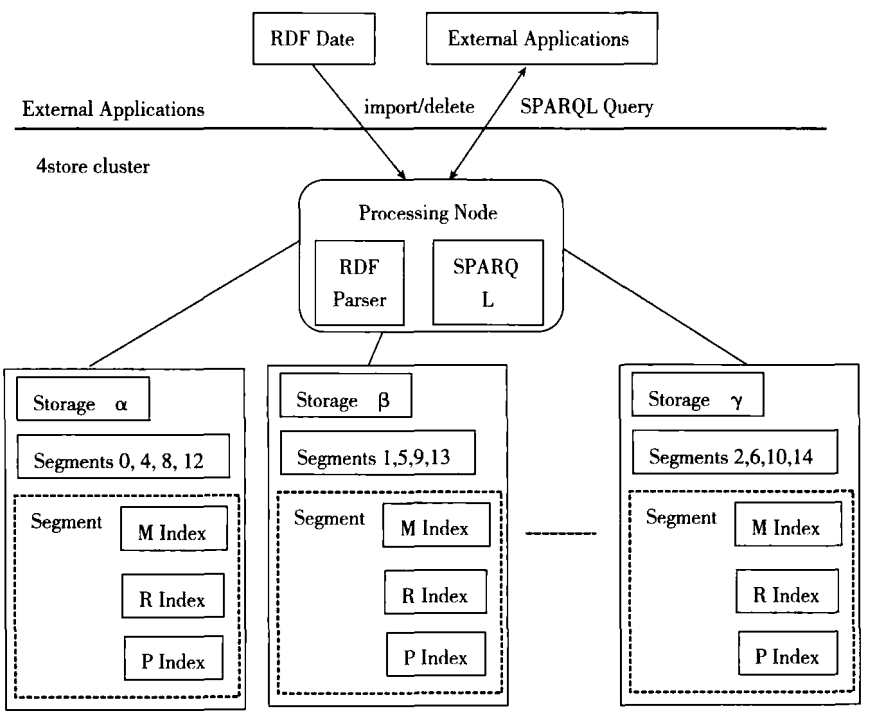

图 2　4store 的系统结构

4.1.3　Bigdata

Bigdata[25]是另一个集群式 RDF 存储系统。它是一个水平扩展的分布式 B + 树数据库系统,支持事务配置、高并发性、高的 I/O 吞吐率。Bigdata 包括了一个高性能的 RDF 数据库系统,可以实现快速加载 RDF 数据和高效率的查询。在存储 RDF 上,它采用 Hash 分区来存储,利用 key - range 划分的 B + 树索引来分布式访问集群上的资源,一个 Bigdata 索引将无符号的 byte[ ]keys 映射到 byte[ ]values,而且这种划分是动态进行的。在查询上,Bigdat 通过集成 Sesame 2 platform 来支持 SPARQL 查询处理。由于 Bigdata RDF 数据库系统能够通过动态的 key - range 分区实现在集群上的分布式操作,因此可以随着 RDF 数据规模的扩大动态的增加节点数目而不用每次都重新加载 RDF 数据到新的节点,从而满足大规模 RDF 数据的存储。

4.1.4　YARS2

YARS2[26]是一个进行过大量优化的集群式 RDF 存储系统。在 RDF 数据存储上,YARS2 采用六个索引来存储 RDF Quad(Subject、Predicate、Object、Context),它使用称为 Sparse Index 的内存数据机构来实现索引,其中的索引项指向了磁盘上每一个 RDF 数据文件的其实位置。在进行查询时,首先在内存中对 Sparse Index 进行二分查找,从而获取所需要的 RDF 数据文件,并对查询的结果采用 ReConRank[27](一种链接分析技术)来进行排序。YARS2 的数据索引和查询过程如图 3 所示。

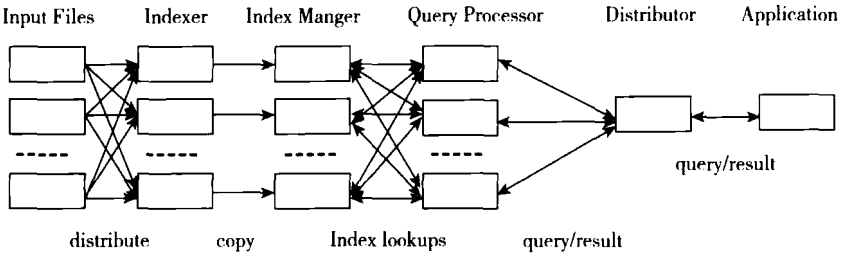

图 3 YARS2 的数据索引和查询过程图

除了上述 4 中分布式 RDF 存储系统外，还有一些类似的分布式 RDF 存储系统，如 Clustered TDB[28]、Virtuoso Cluster Edition[29]等。本文从 RDf 数据存储组织和查询优化以及分布式技术三个方面来对一些分布式 RDF 存储系统进行对比分析，如表 1 所示。

表 1 常见分布式 RDF 存储系统比较

指标 / 系统	存储组织	查询优化	分布式技术
RDFPeers	RDF 三元组按照 S,P,O 被索引三次，存放在不同节点	创建 Native 查询，原子三元组模式、分离和范围查询、多谓语连接查询。	P2P
4Store	RDF Quad 被按照 Subject 划分，再按 M、P、R 索引三次	定义了两个分布式查询操作：bind 和 resolve。	服务器集群
Bigdata	RDF 按照 key - range 划分，使用 B + 树索引	使用 Sesame 2 来实现 SPARQL 查询。	服务器集群
YARS2	使用 Sparse Index 内存索引来索引 RDF 数据	使用 index nested loops join 来实现连接查询。	服务器集群

从表 1 的分析来看，分布式 RDF 存储系统在 RDF 数据存储上，都是采用索引来将 RDF 数据进行划分，从而存储在不同的节点上，实现大规模的 RDF 存储，并且大都采用 hash 来进行索引，不同的是在索引策略上各有不同。在查询优化上，各个系统分别针对各自的存储组织方式进行相应的优化，从而实现查询过程的并行化来提高查询效率。在分布式解决方案上，大都采用服务器集群的方式来实现。其中一些系统也可以部署的单台机器上，如 4store、Bigdata 等。由于 RDFPeers 采用 P2P 技术，因此具有去中心化、可扩展性、健壮性、高性能等特点。而 4store 虽然也具有可扩展性，但是容易受到 Processing Node 的影响，Processing Node 的稳定性和带宽很容易成为系统的瓶颈。

采用传统分布式技术实现的大规模 RDF 存储系统，充分利用了分布式系统的优势，同时具有高可靠性和可用性，可扩展性好，易于集成现有的 RDF 存储系统。同时也存在着一些缺点，比如通信开销大、数据的存取结构复杂而且安全性和保密性难以控制，对于传统的分布式系统，没有专门针对并行处理的优化，使得不能高效查询大规模 RDF 数据。

4.2 采用云计算技术

随着云计算的出现和发展，一些专门处理大数据的工具和解决方案应运而生，其中 Hadoop 是最为广泛使用的。Hadoop 项目中包含了 Google 分布式文件系统、MapReduce 分布式计算框架的开源实现。随着 Hadoopd 的使用，其存储和计算能力得到了认可。因此，很多 RDF 研究人员将目光投向了 Hadoop 上，试图将 Hadoop 和 RDF 相结合，以解决 RDF 的大规模存储和查询问题。这方面的研究目前处于初级阶段，但已经有很多相应的系统被开发出来，我们就一些典型的系统进行分析。

4.2.1 HDFS + MapReduce + RDF

M. F. Husain 等人提出了利用 Hadoop 的 HDFS 来存储 RDF 数据并采用 MapReduce 来处理大规模 RDF 查询的方案[5]。在 RDF 数据存储方面，他们将 RDF 数据以文件的形式存放到 HDFS 分布式文件系统中。

这样存在着一些缺陷,首先是 RDF 数据文件必须经过一些预处理工作才能存放到 HDFS 中,而且直接存放到 HDFS 上,缺乏高效的索引结构。在 RDF 查询处理方面,提出一个贪心的 MapReduce 任务生成算法,多个 MapReduce 任务迭代处理 SPARQL BGP 连接操作,每个任务优先处理共享变量出现次数最多的 Triple Pattern 子句。这种查询响应策略过于简单,不能保证查询的高效性。

4.2.2　HadoopRDF

HadoopRDF[30]是 Yuan Tian 等人提出的基于 Hadoop 和 RDF 存储系统 Sesame 的 RDF 数据存储分析系统。其构想是在 Hadoop 集群的每一个节点上部署一个 Sesame Server 实例,然后通过 Sesame 提供的接口来进行 RDF 数据的存储和查询服务。该系统引入 Hadoop 就是利用 Hadoop 集群来提供高可靠性和高容错恢复能力。同样地,在 RDF 数据查询上,利用 MapReduce 来实现并行化,整个查询过程如图 4 所示。

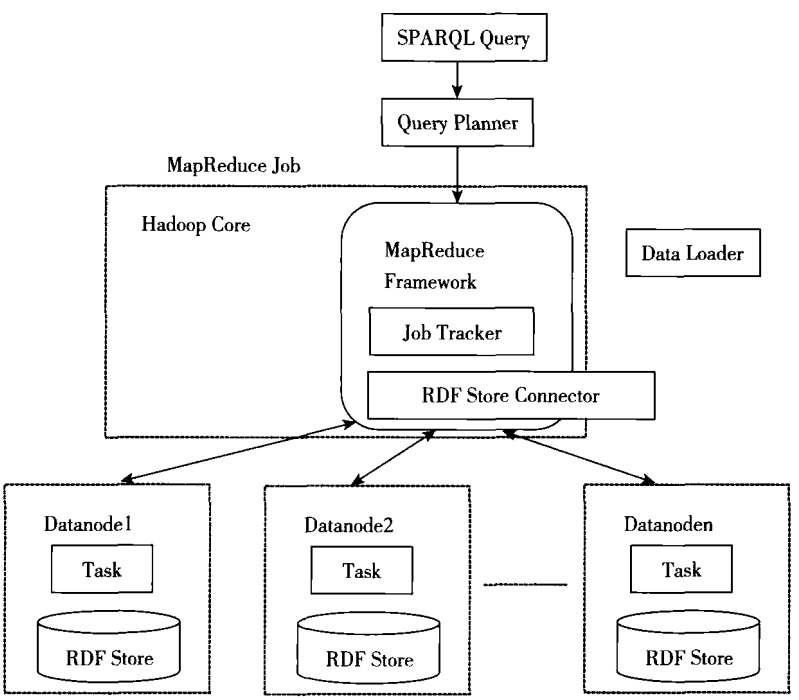

图 4　HadoopRDF 查询执行过程

4.2.3　SPIDER

SPIDER[31]是 Hyunsik Choi 等人提出的基于 Hadoop 的一个大规模 RDF 查询系统。在 RDF 数据存储方面,采用 HBase 数据库进行存储,由于 RDF 是采用三元组来表示数据,而 HBase 是面向列存储稀疏的数据库,因此需要将 RDF 采用面向列的方式进行转换,然后存储到 HBase 上,由于 RDF 数据本身是稀疏的,因此非常适合在 HBase 上存储。在 RDF 查询方面,采用 MapReduce 的方式进行查询,由于每一个节点上只存放部分的 RDF 数据,因此需要 SPIDER 对每个节点的子查询结果进行合并,这个过程会削弱系统的性能,尽管 SPIDER 对 RDF 数据进行了排序,但这种合并造成的损失仍然存在。

4.2.4　H_2RDF

Nikolaos Papailiou 等人提出的 H_2RDF[32]是基于 HBase 和 MapReduce 构建在云上的 RDF 查询系统。H_2RDF 的系统结构如图 5 所示。在数据存储上,对 RDF 数据进行三次索引,分别为 SPO、POS、OSP 三个索引,并且索引以 key - value 的形式存储在 HBase 中。在查询上,从图 5 可以看出,H_2RDF 将查询分为了 MapReduce 查询和 Centralized 查询,对于 SPARQL Query 而言,H_2RDF 首先使用 Jena 进行解析,然后采用连接算法来确定使用 MapReduce 查询还是集中式查询,这样做的结果是,对于简单的查询,可以采用集中式进行,而对于复杂的连接查询,可以使用 MapReduce 进行查询,这样会大大提高查询效率。

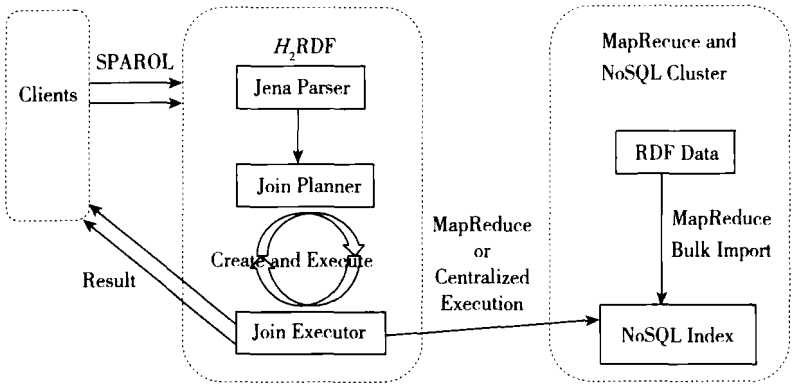

图 5 H_2RDF 系统架构

4.3 综合分析

从上面的几个大规模 RDF 存储系统可以看出,目前的基于云计算的 RDF 存储系统都是建立在 Hadoop 基础上,使用 HDFS、HBase 或其他存储系统来存储大规模 RDF ,采用 MapReduce 来处理查询任务,不同的是各自的存储组织方式和查询优化策略。其中在存储组织上,采用 HDFS 来存储 RDF 数据,这种方法比较简单,不需要额外的转换,缺点是没有高效的索引。而采用 HBase 来存储,可以利用其高效的索引来提高查询效率,但是由于 RDF 数据格式与 HBase 的存储格式不一致,必须进行转换才能存储。在查询上,由于 MapReduce 天然的并行性,因此可以很容易实现查询的并行执行。但是当查询非常简单时,使用 MapReduc 效率反而会降低,因为 MapReduce 需要很多额外的开销。因此对于简单的查询,应该使用集中式查询来实现。

就目前大规模 RDF 存储系统而言,研究主要集中在 RDF 存储组织和查询上,在这两方面已经取得一些成绩,同时也存在着一些问题。

在存储组织上,主要是针对不同存储模式的设计和优化,同时通过建立索引来提高查询效率。这种方式确实提高了查询效率,但是降低了插入或更新时的效率,从目前的大规模 RDF 存储系统来看,大都不考虑插入或更新的效率,理由是对于 RDF 存储系统而言,查询的频繁的,而插入或更新操作很少,一般对于插入或更新操作采取批量处理或离线处理。但是当批量处理的 RDF 数据规模很大时,索引引起的低效率就不容忽视了。

在查询上,主要是针对 SPARQL 查询进行相应的优化,比如采用多级索引或分块索引、预加载、MapReduce 并行化等等。但是现有查询操作都是一次性的,即每次去查询时,都要执行一遍完整的查询过程,客户端必须等待查询结果的返回,这在一定程度上造成了资源浪费。Laurent Pellegrino[33] 等人提出一种基于云的大规模 RDF 数据的 Pub/Sub 查询服务,该服务采用 P2P 和 CEP(Complex Event Processing)技术 ,实现了两种查询模式,即 Pull 模式和 Push 模式。Pull 模式即前面提到的一次性查询,这是阻塞式查询,而 Push 模式采用 Pub/Sub(发布/订阅)的方式为客户端推送结果,这样可以实现异步查询,提高查询效率。

总之,现在大规模 RDF 存储系统都是建立在分布式基础上的,尤其是随着云计算的不断发展和完善,会有更多的分布式 RDF 存储系统和基于 Hadoop 的 RDF 存储系统被开发。虽然已经有很多的分布式 RDF 存储系统,但是都各自为战,没有统一的标准和接口,目前为止,尚未有一种通用的大规模 RDF 存储系统。

5 总结与展望

本文在充分调研和深入分析的基础上对大规模的 RDF 数据存储技术进行了综述。其中重点介绍了 RDF 存储系统的组织方式,包括基于内存的存储、基于文件系统的存储和基于关系型数据库的存储,同时重点介绍了大规模 RDF 存储系统的解决方案以及现有的一些系统的比较和分析,虽然大规模 RDF 存储系统的研究还处于初级阶段,但是已经有很多研究机构在进行研究,在本文的最后,基于目前已有的一些研究经验提出一些问题以及值得进一步探究的研究点。

首先目前对于大规模 RDF 存储系统的研究主要是基于 Hadoop 之上,采用 HDFS 或 HBase 来存储 RDF 数据,采用 MapReduce 来实现分布式查询,但是 RDF 数据格式与 HBase 数据格式的不一致,导致必须进行转换,是否可以设计一种直接存储 RDF 格式数据的存储方式,这种存储方式同时可以满足类似 MapReduce 的并行化查询,这样就可以跳过转换步骤,直接进行存储和查询。其次,目前关于大规模 RDF 存储系统的研究比较多,但是没有一种通用的架构或解决方案被提出来,已有的系统都是各自为战,以后的移植都是一个问题,今后可以就这方面研究一种比较通用的解决方案或架构等。最后,由于目前的大规模 RDF 存储系统都是基于云计算的,而云计算是一种集中式计算然后提供分布式服务,这种集中式计算容易造成资源浪费和负载不均衡,尽管云计算架构中有成千上万个节点,随着数据量的增加,负责计算的节点很可能成为系统的瓶颈,未来,可以考虑云计算和 P2P 相结合,充分发挥云计算的计算能力和 P2P 的去中心化、高可靠性的特点,来满足更大规模的 RDF 存储。

参 考 文 献

[1]http://www.w3.org/standards/semanticweb. World - Wide Web Consortium: Semantic Web

[2]http://www.w3.org/RDF. World - Wide Web Consortium: Resource Description Framework.

[3]Y. Theoharis, V. Christophides, and G. Karvounarakis, Benchmarking Database Representations of RDF/S Stores, In Proceedings of the 4th International Semantic Web Conference (ISWC'05),2005.688 - 700

[4]M. Cai and M. R. Frank. RDFPeers:a scalable distributed RDF repository based on a structured peer - to - peer network. In Proceedings of the 13th International conference on World Wide Web(www'04),2004.652 - 654

[5]M. F. Husain, P. Doshi, L. Khan, and B. Thuraisingham, Storage and Retrieval of Large RDF Graph Using Hadoop and MapReduce, CloudCom 2009.680 - 686

[6]J. H. Du, H. F. Wang, Y. Ni, and Y. Yu, HadoopRDF: A Scalable Semantic Data Analytical Engine , ICIC 2012.633 - 641

[7]Berners - Lee T, Hendler J, Lassila O. The semantic Web. Scientific American, May 2001,1 - 2

[8]Berners - Lee T. Artificial intelligence and the semantic Web. In: Proc. of the AAAI 2006 Keynote. 2006,7 - 14

[9]Fikes R, Hayes P, Horrocks I. OWL - QL: A language for deductive query answering on the semantic Web. Journal of Web Semantics, 2004,2(1):19 - 29

[10]Karvounarakis G, Alexaki S, Christophides V. RQL: A declarative query language for RDF. In: Lassner D, Roure DD, Iyengar A,eds. Proc. of the WWW 2002. New York: ACM Press, 2002.592 - 603

[11]Seaborne A. RDQL - A query language for RDF. W3C, 2004 01 09, 2004. http://www.w3.org/Submission/RDQL/.

[12]Prud'hommeaux E, Seaborne A. SPARQL query language for RDF. W3C,2008 0115, 2008. http://www.w3.org/TR/rdf - sparql - query/

[13]鲍文,李冠宇. 本体存储管理技术研究综述. 中国科技论文在线, 2007.04.4 - 6

[14] J. Broekstra, A. Kampman, F. Harmelen. Sesame: A Generic Architecture for Storing and Querying RDF and RDF Schema[A]. In Proc. of the 1st International Semantic Web Conference[C]. Sardinia, Italy, June 2002.54 - 68

[15]常冰琳,袁平鹏. 面向在线分析的语义网数据存储系统研究.2011,(S2):17 - 22

[16]David Wood, Paul Gearon, Tom Adams. Kowari: A Platform for Semantic Web Storage and Analysis. In Proc. of WWW 2005. Chiba, Japan, May 2005.

[17]Wu G, Li JZ, Hu JQ et al. System Π: A native RDF repository based on the hypergraph representation for RDF data model[J]. JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY, 2009,24(4):2 - 5

[18]Jeremy J. Carroll, Dave Reynolds, Ian Dickinson, et al. Jena: Implementing the Semantic Web Recommendations. In Proc. of WWW 2004, May 2004.2 - 7

[19]S. Harris and N. Gibbins. 3store: Efficient Bulk RDF Storage[A]. In Proc. of the 1st International Workshop on Practical and Scalable Semantic Systems[C]. Sanibel Island, Florida, USA, 2003. 1 - 15

[20]Li Ma, Zhong Su, Yue Pan, et al. RStar: An RDF Storage and Query System for Enterprise Resource Management[A]. In Proc of CIKM 2004[C]. New York, NY, USA, 2004. 484 - 491

[21] Agrawal R, Somani A, Xu Y. Storage and querying of e - commerce data. In: Peter M G A, Paolo A, eds. Proc of the 27th VLDB. Roma: Morgan Kaufman Piblishers Inc, 2001. 149 - 158

[22] Brian McBride. Jena: implementing the RDF model and syntax specification, Technical Report, Hewlett Packard Laboratories, Bristol, 2000. 1 - 4

[23] Pan Z X, Heflin J. DLDB: extending relational database to support semantic Web queries. In: Raphael Vclz, Stefan Decker, et al eds. Proc of the 1st PASS. Santa Barbara: Informal Proceedings, 2003. 2 - 4

[24] S. Harris, N. Lamb, and N. Shadbolt. 4store: The Design and Implementation of a Clustered RDF Store. InProceedings of the 8th International Semantic Web Conference (ISWC'09), 2009. 81 - 95

[25] Personick, M. : Bigdata: Approaching web scale for the semantic, http://www. bigdata. com /bigdata_whitepaper_07 - 08 - 2009. pdf, 2009.

[26] A. Harth, J. Umbrich, A. Hogan, and S. Decker. YARS2: A Federated Repository for Querying Graph Structured Data from the Web. In Proceedings of the 6th International Semantic Web Conference (ISWC07), 2007.

[27] A. Hogan, A. Harth, and S. Decker. ReConRank: A Scalable Ranking Method for Semantic Web Data with Context. In 2nd Workshop on Scalable Semantic Web Knowledge Base Systems, 2006.

[28] A. Owens, A. Seaborne, and N. Gibbins. Clustered TDB: A Clustered Triple Store for Jena. In Proceedings of the 18th International Conference on World Wide Web (www09), 2009.

[29] Erling , O. Mikhailov, Towards web scale RDF. In: Proceedings of the 4th International Workshop on Scalable Semantic Web Knowledge. October 2008.

[30] Y. Tian, J. Du, H. F. Wang, and Y. Yu. HadoopRDF: A Scalable RDF Data Analysis System. Submissions of Semantic Web Chanllenge, 2010.

[31] H. Choi, J. Son, Y. H. Cho, etc, SPIDER: A System for Scalable, Parallel/Distributed Evaluation of large - scale RDF Data, In Proceedings of the 18th ACM Conference on Information and Knowledge Management (CIKM09), 2009.

[32] N. Papailiou, L. Konstantinou, D. Tsoumakos, and N. Koziris, H2RDF: Adaptive Query Processing on RDF Data in the Cloud, WWW 2012.

[33] L. Pellegrino, F. Baude, I. Alshabani, Towards a Scalable Cloud - based RDF Storage Offering a Pub/Sub Query Service, CLOUD COMPUTING 2012.

作者简介

何少鹏，男，1990 年生，硕士研究生，主要研究方向是科学数据的发布与语义搜索。

黎建辉，男，1973 年生，研究员，主要研究方向是云计算、大数据处理、大规模数据的组织与集成。

沈志宏，男，1977 年生，高级工程师，主要研究方向是科学数据的组织、集成与管理。

周园春，男，1975 年生，副研究员，主要研究方向是数据挖掘、大数据处理。